http://dx.doi.org/10.5516/NET.04.2012.061

A DEVELOPMENT FRAMEWORK FOR SOFTWARE SECURITY IN NUCLEAR SAFETY SYSTEMS: INTEGRATING SECURE DEVELOPMENT AND SYSTEM SECURITY ACTIVITIES

JAEKWAN PARK* and YONGSUK SUH
Korea Atomic Energy Research Institute
Daedeok-daero 989-111, Dukjin-dong,Yuseong-gu, Daejeon, Korea
Corresponding author. E-mail : jkpark183@kaeri.re.kr

Received September 07, 2012
Accepted for Publication June 13, 2013

The protection of nuclear safety software is essential in that a failure can result in significant economic loss and physical damage to the public. However, software security has often been ignored in nuclear safety software development. To enforce security considerations, nuclear regulator commission recently issued and revised the security regulations for nuclear computer-based systems. It is a great challenge for nuclear developers to comply with the security requirements. However, there is still no clear software development process regarding security activities. This paper proposes an integrated development process suitable for the secure development requirements and system security requirements described by various regulatory bodies. It provides a three-stage framework with eight security activities as the software development process. Detailed descriptions are useful for software developers and licensees to understand the regulatory requirements and to establish a detailed activity plan for software design and engineering.

KEYWORDS : Nuclear Safety Software, Nuclear Software Development, Software Development Process, Software Security, Secure System

1. INTRODUCTION

Many nuclear power plants have recently been upgraded from analogue-based manual systems to computer-based control systems. The control systems perform data acquisition, control actuation, and information indication based on software. Existing operator actions, such as the monitoring of hardwired panels and the manual control of hand switches, have been replaced with computer-based visualization and automatic actuation. It also supports faster responses in plant operation and reduces the human resources and costs. However, there are a few disadvantages induced by computer-based systems. One of the most severe disadvantages is the security vulnerability of Ethernet-based communication and a rise in software threats.

The main differences between nuclear specific software and general purpose software are the safety functions in that a failure can result in significant economic loss and physical damage to the public. The design and development processes of safety functions have been established, guided, and regulated by the safety regulations and standards in the nuclear industry, such as 10 CFR 55, IEEE Std. 279-1971, IEEE Std. 603-1991, and IEEE Std. 7-4.3.2-2003. In terms of security, the U.S. Nuclear Regulatory Commission (USNRC) revised Regulatory Guide (RG) 1.152-2011 [1], which provides specific system features and development activity guidance concerning the security of computer-based safety systems. In addition, USNRC issued RG. 5.71-2010 [2], which provides security functions and activities for establishing and maintaining security capability. However, these regulatory guidelines focus on different aspects, i.e., secure development activity and system security activity, and are not specific enough for a systematic treatment of safety systems.

There are many studies on enhancing the reliability and safety of critical software [3-5] and addressing the security problem [6, 7] in the software development process. Currently, the most important part of the software development process for the achievement of safety system security is security engineering, which describes how to integrate security activities into a software lifecycle model. Recently, Chou[8] proposed a regulatory-based development process that describes the specific development stages for the software security of safety systems. It is a well-structured process based on RG. 1.152-2006 [9]. A new development process that integrates both RG. 5.71-2010 and RG. 1.152 -2011 is required.

This paper is organized as follows. The history of cyber security accidents and efforts are presented in Section 2. Section 3 introduces the relevant nuclear regulations and regulatory guidelines of software security. In Section 4, an integrated development process is proposed for the software security of safety systems. Finally, some conclusions and future work are given in Section 5.

2. CYBER SECURITY ACCIDENTS AND EFFORTS

It was recently reported that several plants have been attacked and damaged by outside intruders [10]. On January, 2003, the Slammer worm attacked a vulnerability in the systems at the Davis-Besse nuclear power plant, and the computer systems and safety parameter display systems were infected. Because of network traffic generated by the worm, the plant personnel could not access the safety parameter display systems. In August, 2006, a shutdown of Unit 3 at the Browns Ferry nuclear power plant showed that even critical reactor components can be disrupted and disabled by a cyber attack. Unit 3 was manually shut down after the failure of the controllers with embedded microprocessors and Ethernet communication capabilities. On July, 2010, the Stuxnet worm was detected at the Bushehr nuclear power plant. The worm exploited a vulnerability in Microsoft Windows to infect systems adopting Siemens control software.

To cope with these cyber attacks, various studies have been carried out in the IT and nuclear industries. Zakaria I. Saleh[11] suggested a security risk assessment framework in the IT industry. The framework includes processes for a security risk and vulnerability assessment. As efforts in the nuclear industries, Nai Fovino I[12] presented the outcome of information and communication technology (ICT) security assessment targeting an operational power plant. The results show that the vulnerability of a plant to malicious attacks is severe. Lee[13] introduced a practice for a cyber security risk assessment in power plants as required by RG. 1.152-2006. The assessment consists of a target system analysis, asset analysis, threat analysis, vulnerability analysis, risk analysis, and intrusion tests to identify the risks.

As emphasized in previous studies, safety systems should be strengthened against unauthorized accesses, and security controls should be employed. To address these issues, national laboratories, utilities, and regulatory bodies have tried for a long time to find the best way to cope with not only attacks by intruders from outside but sabotage from inside. Since 2006, regulatory guidelines and industry standards for cyber security have been published. Therefore, these guidelines should be strongly considered in the development process of digital systems used in nuclear plants.

However, there are few studies on software development frameworks concerning the emerging cyber security issues. Most digital safety systems have been developed under the software development process guided by the industrial standard (e.g., IEEE 7-4.3.2-2003) without cyber security considerations. As a software development model, Chou[8] proposed a regulatory-based development method to meet the security requirements of the regulatory guidelines, RG. 1.152-2006. This model, based on the traditional software development life cycle, includes additional activities for the security requirements in the development and operation phases. Recently, the existing security requirements for the development phase have been changed and several security requirements for the operation phase have been newly established. Thus, the previous method does not completely satisfy the requirements of the current regulations. This paper proposes an integrated development process suitable for both secure development requirements and newly updated system security requirements. It includes appropriate activities to meet the up-to-date requirements, such as a secure development environment, defense-in-depth, digital asset analysis, and security assessments.

3. CYBER SECURITY REQUIREMENTS

In 2011, USNRC issued a new revision of RG. 1.152 -2011, "Criteria for use of computers in safety systems of nuclear power plants," which uses the waterfall life-cycle phases as a framework to describe the system security guidance. The framework consists of five phases: the concept, requirement, design, implementation, and test phases. The abstract contents of this guidance are listed in Table 1.

In 2010, USNRC issued new guidelines, RG. 5.71-2010, "Cyber security programs for nuclear facilities", which describes the technical methods and security activities for the operation and maintenance of a nuclear plant. Cyber security features should be designed and implemented during the development phase before the site installation of the systems, as any later treatment of the systems for security may cause unpredicted defects in the systems or may be implemented with less effective security measures. This means that security controls concerned in RG. 5.71-2010 should also be planned, designed, and implemented during the safety system development phase. However, it does not provide specific lifecycle-based processes. The main activities are summarized in Table 2.

To achieve conformance with both types of guidance, it is necessary to infuse the security requirements of RG. 1.152-2011 and RG. 5.71-2010 into every stage of the system lifecycle. We integrated the five sections of RG. 1.152-2011 and the main activities of RG. 5.71 into three stages: planning stage, development stage, and operation and maintenance phase. Before going any further, we will explain briefly the RG. 5.71-2010 and RG. 1.152-2011 viewpoints for software security below:

(1) High potential threats from information technology (IT):
There is no disagreement that information technology

Table 1. Summary of RG. 1.152-2011

Sections	Descriptions
Concept	* Establish a secure operational environment
	* Identify potential security vulnerabilities
	* Remote access should not be implemented
Requirements	* Define security functional requirements
	* V&V role
	* Pre-developed software should be addressed
	* Secure development process
Design	* Specific design configuration items
	* Developer should take the standards and procedures
Implementation	* Implement security procedures and standards
	* Testing to address undocumented codes
Test	* Verify security functions
	* Test should cover overall system

Table 2. Summary of RG. 5.71-2010

Sections	Descriptions
Cyber Security Program Establishment	* Cyber security team, training plan
	* Critical digital assets analysis
	* Defense-in-depth strategy
	* Implement security controls
Cyber Security Program Maintaining	* Continuous monitoring
	* Periodic assessment and audit
	* Change control
	* Cyber security program review

is essential in the world today, and terrorism constitutes a major threat to the IT industry. Nevertheless, nuclear safety systems rely on IT, such as Commercial Off-The-Shelf (COTS) products and Ethernet networks for data communication and process control. Indeed, the nuclear industry also faces a risk of terrorism.

(2) Seamlessly addressing the security considerations of digital safety systems: A combination of RG. 1.152-2011 and RG. 5.71-2010 can seamlessly address the secure design, development, and operation of digital safety systems. The former addresses the security issues during safety system development, and the latter provides programmatic security guidance for operation and maintenance.

(3) The main issues of RG. 1.152-2011: The safety system design for a secure operational environment should address the physical and logical access to the system functions, the use of safety system services, and data communication with other systems. In addition, standards and procedures should be implemented for a secure development environment.

(4) The main elements of RG. 5.71-2010: A defense-in-depth strategy is an activity to establish multiple layers of protection to guard safety systems containing critical digital assets (CDAs), as the failure of a single layer should not result in a compromise of CDAs. The application of security controls classified as technical controls, operational controls, and management controls is also an important activity that makes up safeguard or protective measures addressing the potential cyber risks of CDAs. Continuous monitoring of their effectiveness is also a critical activity during the plant operation stage.

4. AN INTEGRATED DEVELOPMENT PROCESS

In this section, we propose a three-stage development process approach based on a security regulation analysis. Within this process, we also propose eight security activities, which are grouped into three stages as shown in Fig 1. Each security activity corresponds to a phase of the software development lifecycle.

4.1 Planning Stage

Security conceptualization is the main activity at this stage. Its goals are to establish a security policy, identify security capabilities based on this policy, and prepare

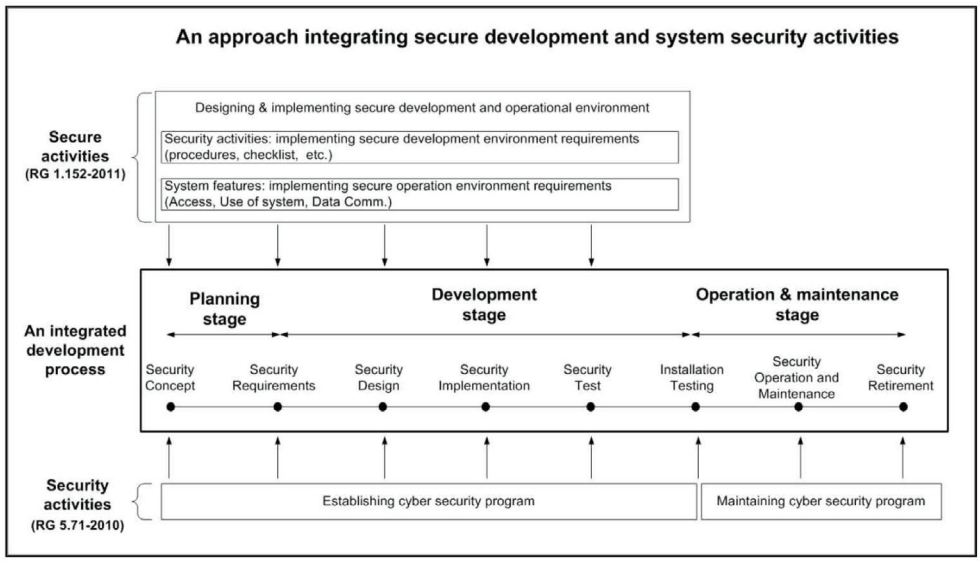

Fig. 1. Process Flow Diagram of the AFC Facility

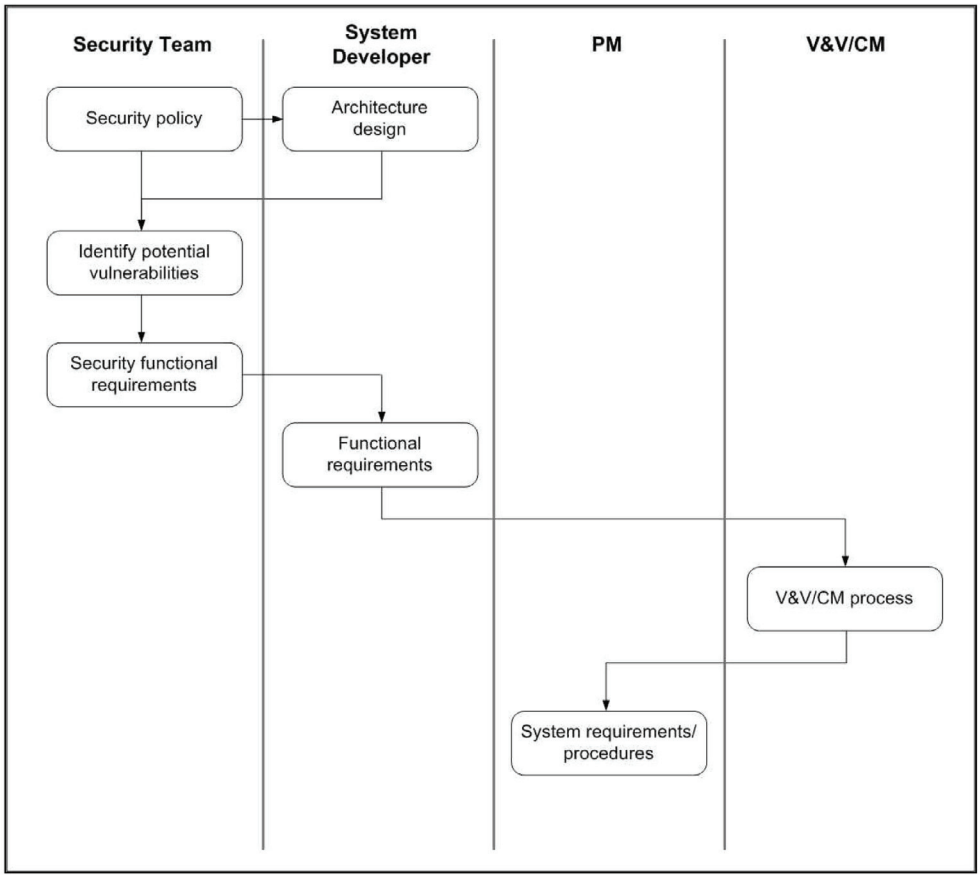

Fig. 2. Activity Diagram of the Planning Stage

security specifications. Some security constraints should be included in the security policy. For example, remote access should not be allowed, data communication from safety systems to non-safety systems should have one-way communication pathways, and so on. The constraints are provided to the developers as a requirement in the architecture design. We describe the sequence of activities in this stage, as shown in Fig. 2.

(1) As top-level security requirements, the security team establishes a security policy to limit the scope of the system security. The policy is referred to in order to analyze the security requirements of safety systems. In addition, the security team performs a security assessment to identify potential vulnerabilities based on the architecture design. The risk of software intrusion must be mitigated by introducing security functions as countermeasures. These functional requirements may include well-known security requirements, such as access control, one-way communication pathways to non-safety systems, highly reliable modification procedures, and exclusion of remote access. The functional requirements of security are incorporated into the system requirements.
(2) Quality assurance (QA) activities are supported by associated teams. The V&V team verifies the correctness and completeness of the software security and the overall software requirements. In addition, the configuration management (CM) team defines the security configuration items as a part of the system configuration items.
(3) The project management (PM) team is in charge of reviewing and approving the safety system requirements including the security requirements. The team should also prepare standard development procedures to prevent the introduction of unwanted or unnecessary functions and codes during the development process. That is, the main outputs at this stage are software functional requirements and security-related procedures.

4.2 Development Stage

Based on the results of the planning stage, there are five security activities (see Fig. 1) to be performed at this stage.

4.2.1 Requirements Analysis

The results of the planning stage should be incorporated into software requirements fundamentally. At this stage, additional requirement analyses are conducted. The cyber security team should perform an analysis to identify the critical digital assets based on the functional requirements and the architecture design. The critical digital assets should be protected strongly using additional security controls. Generally, most digital assets performing safety functions are classified as critical digital assets. Furthermore, the security team should establish a defensive architecture with multiple layers of protection to safeguard the critical digital assets. Its purpose is to ensure that the failure of a non-critical asset does not result in the compromise of critical assets. For example, the critical assets are located at the lowest security level and others are appropriately located at higher security levels. Boundary security controls, e.g., a firewall, are employed for screening malicious communications from non-critical assets to critical assets.

In addition to the above requirements, the specific security requirements are listed below:

(1) Pre-developed software requirements: COTS and reused software should address the vulnerability of the safety software by using software functions that have been tested and are supported by operating experience.
(2) Access control requirement: a combination of software (e.g., password) and hardware (e.g., key, smart-card) is needed rather than just a password.
(3) Interface requirement: only one-way communication is allowed for transferring data from safety systems to other systems, a remote access point to a safety system is not allowed, and a cryptography mechanism is implemented for data transmission and information integrity during the use of Ethernet-based communication.
(4) Operation and maintenance requirement: a periodic security monitoring and assessment should be planned and performed, and an incident response plan should be proposed during the operation and maintenance phase.
(5) Retirement requirement: the effect of replacing or removing existing safety system security functions should be assessed during the decommissioning period.

4.2.2 Development Processes

Based on the above results of the requirement analysis, four activities should be performed. We proposed an activity diagram to represent the activities and processes shown in Fig. 3:

(1) The developers should define the security configuration items and translate them into system specifications. The V&V/CM team should identify the security configuration items and verify the security requirements based on the system specifications.
(2) Next, developers incorporate the specific configuration items into a software design description. The description should address control over (a) physical and logical access to the software functions, (b) the use of safety software services, (3) data communication with other systems, and (d) a defense-in-depth architecture. Moreover, the development team should follow the developer's guideline to avoid the introduction of undocumented codes, malicious codes, and other unwanted or undocumented functions or applications. The developer guidelines or procedures contain a self-checklist for software design principles, such as accuracy, clarity, loose coupling, and strong cohesion [14]. The security team should review whether the security requirements are mapped into the appropriate design items. The V&V and CM teams are also in charge of the verification and change control of the security configuration items, respectively.
(3) During the implementation phase, the development team transforms the software design into code, database structures, and related machine executable representations. They may need static analysis tools to detect common vulnerabilities, implementation flaws, and

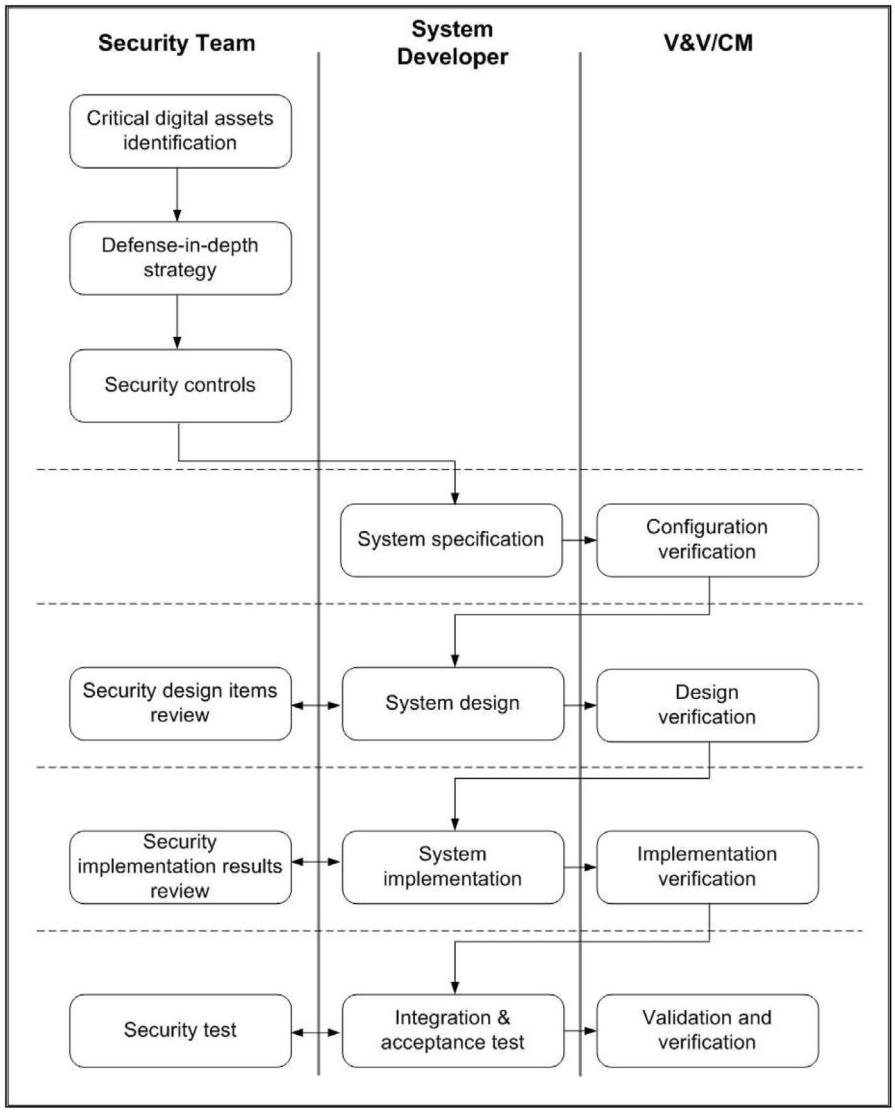

Fig. 3. Activity Diagram of the Development Stage

source code bugs. An independent code review may also be used to achieve a more secure software implementation. The security team should review whether the security design items are implemented well. The V&V team should ensure that the design transformation is correct, accurate, and complete. The CM team focuses on the change control of the security configuration items.

(4) During the factory acceptance test and installation phase, the security team and developers should conduct hardware configuration, integration, qualification, factory acceptance, and installation tests to verify the software security features. Conducting security tests or drills is useful to identify the actual security capability of the system. The V&V team should be in charge of the verification and validation of the overall software testing. In addition, they should also ensure the correctness of the safety software security features in the target environment after the software is shipped to the nuclear power plant.

4.3 Operation and Maintenance Stage

The final stage (see Fig. 1) is performed by three security activities including security operation, security maintenance, and security decommissioning. These activities and processes are described in the activity diagram shown in Fig. 4.

(1) During normal operation, the licensee establishes and maintains a site security program that includes periodic testing and monitoring, a review of software logs, and real-time monitoring. The security team should develop an incident response and recovery plan for responding to digital system security incidents.

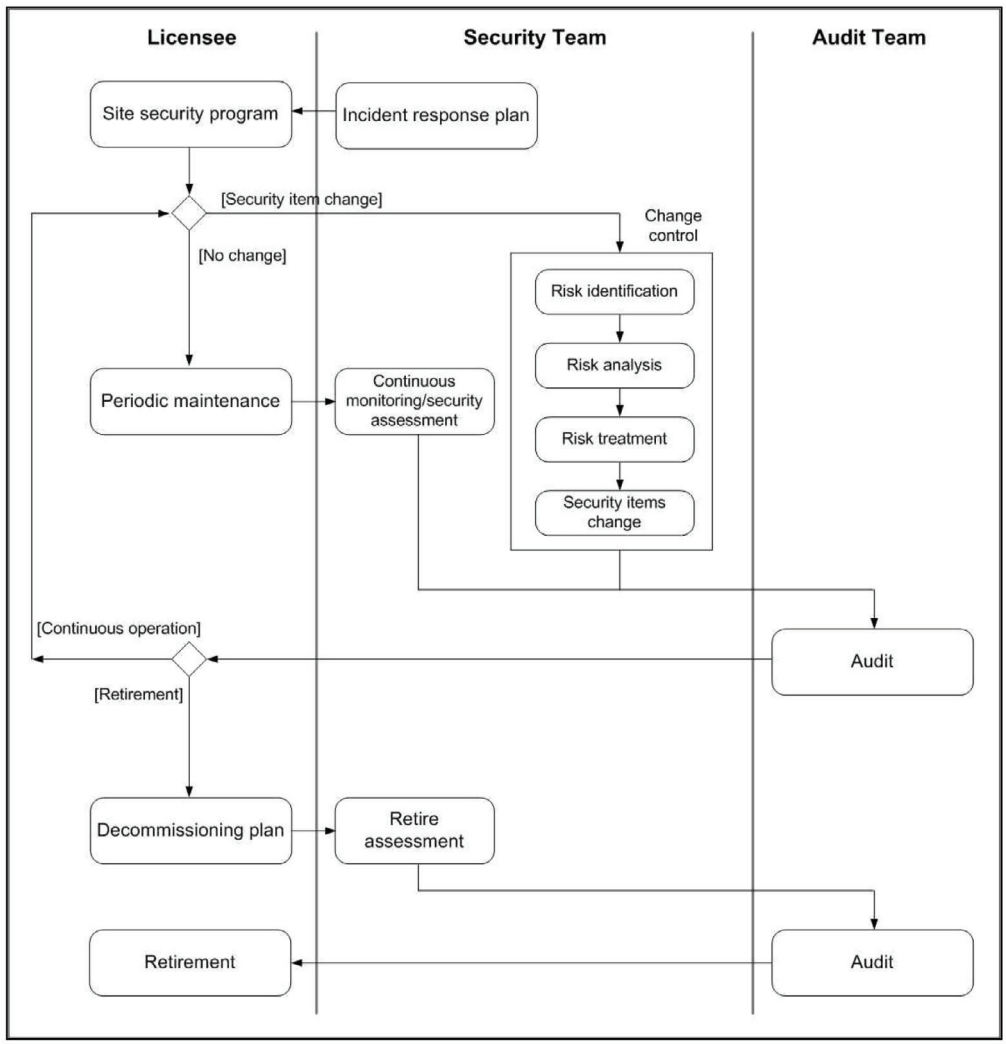

Fig. 4. Activity Diagram of the Operation and Maintenance Stage

(2) If any change is proposed by the licensee, the security team then identifies the potential threat induced by the item replacement and change in the operating environment. In addition, the team performs a risk analysis to evaluate the impact of safety software changes in the operating environment. The risk analysis should include a data flow diagram analysis, dependency analysis, and interface analysis. When a new potential threat impacts the safety system, additional security measures for mitigating the vulnerability are recommended in the risk treatment step. If there is no change, the security team performs a continuous monitoring and security assessment periodically. The audit team reviews the security program and related activities to ensure that vulnerabilities are not introduced into the plant environment.

(3) During the retirement phase, the licensee proposes a decommission plan that contains the methods by which a change in the safety software security functions will be mitigated. The security team should assess the effect of replacing or removing the existing safety system security functions from the operating environment. Additionally, the audit team should review the assessment results. Upon removal from service, the licensee should consider data cleansing, disk destruction, or a complete overwrite.

5. CONCLUSION

As digital technologies have been adopted in the development of nuclear safety systems, security has also become an important issue for the nuclear industry. However, security has rarely been considered in the software development of safety systems. Facing new security challenges, USNRC issued several regulatory guidelines concerning computer-

based safety system security. However, there is still no clear development process regarding security activities in the software development lifecycle.

This paper proposes an integrated development process that combines the security activities of the major regulatory guidelines, RG. 1.152-2011 and RG. 5.71-2010. The contributions of this paper are as follows:

- Our proposed approach is a comprehensive development process based on both the secure development regulations and the security regulations for nuclear safety software. It provides a three-stage framework with eight security activities in legacy software project management. It can be used to address the security requirements early during the software development process.
- This approach emphasizes software design and engineering for meeting nuclear regulation requirements. For example, remote access to the software design of safety systems should not be allowed.
- Detailed descriptions in this paper are useful for software developers and licensees to better understand the regulatory requirements.

The proposed framework needs more effort than the previous software development methods because many security activities are added to the traditional software life cycle model. However, this can lead to safer operation of digital safety systems and in any case it cannot be omitted due to the current licensing requirements of regulatory bodies. Compared with previous software development methods, it enhances the safety of system by maintaining the system integrity more safely against various cyber threats. For example, the defense-in-depth architecture can make it hard to directly attack the safety system, and the security controls of the security boundaries and safety systems can prevent the attacks, and the security monitoring and assessment can minimize vulnerabilities introduced into the plant environment.

We will evaluate the engineering processes in further research. We also anticipate that merging an asset analysis and design assessment will be the direction of our future research.

REFERENCES

[1] USNRC. Regulatory Guide 1.152 Revision 3, "Criteria for Use of Computers in Safety Systems of Nuclear Power Plants", 2011.

[2] USNRC. Regulatory Guide 5.71, "Cyber Security Programs for Nuclear Facilities", 2010.

[3] Ghahramani B., "Software reliability analysis: a systems development model", *Computers & Industrial Engineering*, vol. 45, pp. 295-305 (2003).

[4] Weber W, Tondok H, Bachmayer M, "Enhancing software safety by fault trees: experiences from an application to flight critical software", *Reliability Engineering & System Safety*, vol. 89, pp. 57-70 (2005).

[5] Nai Fovino I, Masera M, De Cian A, "Integrating cyber attacks within fault trees", *Reliability Engineering & System Safety*, vol. 94, pp. 1394-1402 (2009).

[6] Chou IH, "Secure Software Configuration Management Processes for nuclear safety software development environment", *Annals of Nuclear Energy*, vol. 38, pp. 2174-2179 (2011).

[7] Lahtinen J, Valkonen J, Björkman K, Frits J, Niemelä I, Heljanko K, "Model checking of safety-critical software in the nuclear engineering domain", *Reliability Engineering & System Safety*, vol. 105, pp. 104-113 (2012).

[8] Chou IH, Fan C-F, "Regulatory-based development processes for software security in nuclear safety systems", *Progress in Nuclear Energy*, vol. 52, pp. 395-402 (2010).

[9] USNRC. Regulatory Guide 1.152 Revision 2, "Criteria for Use of Computers in Safety Systems of Nuclear Power Plants", 2006.

[10] Kesler B, "The Vulnerability of Nuclear Facilities to Cyber Attack", *Strategic Insights*, vol. 10, pp. 15-25 (2011).

[11] Zakaria I. Saleh, Heba Refai, Mashhour A, "Proposed Framework for Security Risk Assessment", *Journal of Information Security*, vol. 2, pp. 85-90 (2011).

[12] Nai Fovino I, Guidi L, Masera M, Stefanini A, "Cyber security assessment of a power plant", *Electric Power Systems Research*, vol. 81, pp. 518-526 (2011).

[13] Lee CK, Park GY, Kwon KC, Hahn DH, Cho SH, "Cyber security design requirements based on a risk assessment", *Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies*, pp. 1638-1646 (2009).

[14] Mark D, John MD, Justin S., "The art of software security assessment", Addison-Wesley (2007).